# Reference Usage Manual for SmartLearn Kids Assignment

This manual shows where each reference was used in the SmartLearn Kids Software Engineering assignment, including chapter and page guidance where applicable. It also includes explanations for key concepts such as Black-box testing, MoSCoW prioritization, abstraction, and modularity in the context of the project. This is for easier understanding and verification.

## 1. Sommerville, I. (2016). Software Engineering (10th ed.). Pearson

Type: Book

Accessibility: Not freely available; some versions may be accessible via Internet Archive.

Usage in Report with Chapters/Pages:

**Q2:** Proposed Product Specification (Chapter 4, Requirements Engineering)  
**Q5:** Software Process Models and Agile (Chapter 3 P73-74), Agile software development)  
**Q7:** Version Control Strategy (Chapter 25 P(735-739), Configuration Management)  
**Q8:** Software Quality Measures (Chapter 24, Quality Management)  
**Q9:** Testing/Validation Strategies (Chapter 8 P(236), Software Testing)

Key Concepts Explanation:

* Black-box testing: Testing functionality without considering internal code. Example: Enter numbers in a quiz, check the score, don’t need to see the scoring function code.

## 2. Rwanda Education Board (REB) Curriculum Documents, 2023

Type: Official Curriculum Document

Accessibility: Freely accessible online

Usage in Report with guidance:

* Q1: Problem Statement (Curriculum objectives)
* Q2: Product Specification, Subjects & Levels
* Q3: Preliminary Requirements Analysis (Grade-appropriate content)

## 3. GASHEMA, G. (2025). Lecture 3: Software Process and Software Process Models

Type: Lecture Notes

Usage in Report with guidance:

* Q5: Agile Methodology explanation and reasoning (Lecture3: Agile Development, slide 26-27)
* Q6: Overall Plan (Lecture2: SDLC Phases, Slide-8)
* Q8:Software Quality Measures (Lecture1: What are the attributes of good software? (Slide-41)

## 4. UNIT‑III: Design Concepts and Abstraction

Type: Online Resource

Accessibility: Not Freely accessible online

Usage in Report with guidance:

**Q6:** Abstraction and Modularity

Key Concepts Explanation:

* Abstraction: Focusing on essential features while hiding implementation details. Example: Students only see the video interface, not how lessons are loaded from the database.
* Modularity: Dividing the system into independent modules for easier maintenance and development. Example: Separate modules for UserManagement, CourseModule, and ProgressTracking.

Citation:

UNIT‑III. (n.d.). UNIT III: Design Concepts and Abstraction [Document]. Scribd. Retrieved October 8, 2025, from [https://www.scribd.com/document/741583673/UNIT-III](https://www.scribd.com/document/741583673/UNIT-III?utm_source=chatgpt.com)

## 5. MOSCOW Prioritization Tech (Scribd)

Type: Online Resource

Accessibility: Freely accessible online

Usage in Report with guidance:

* Q3: MoSCoW Prioritization (practical application explanation)

Key Concepts Explanation:

* MoSCoW prioritization: Technique for classifying requirements into Must/Should/Could/Won’t have.

Citation:  
Sundar\_nycusa. (n.d.). MOSCOW Prioritization Tech. Scribd. Retrieved October 8, 2025, from https://www.scribd.com/document/909469860/MOSCOW-Prioritization-Tech

## 6. Version Control Systems (VCS) Paper (ResearchGate)

Type: Online Research Paper

Accessibility: Freely accessible via ResearchGate

Usage in Report with guidance:

* Q7: Version/Revision Control Strategy (discussion of Git, GitHub, and best practices)

Citation:  
Devineni, S. K. (2020). Version Control Systems (VCS): The Pillars of Modern Software Development – Analyzing the Past, Present, and Anticipating Future Trends. International Journal of Science and Research (IJSR), 9(12), 1816–1829.  
Retrieved from https://www.researchgate.net/publication/378490782\_Version\_Control\_Systems\_VCS\_the\_Pillars\_of\_Modern\_Software\_Development\_Analyzing\_the\_Past\_Present\_and\_Anticipating\_Future\_Trends